# Up close, but not too personal
## Hypotargeting for recommender systems

Larson, Martha; Slokom, Manel

**Publication date**
2019

**Document Version**
Final published version

**Published in**
ImpactRS 2019 Impact of Recommender Systems 2019

**Citation (APA)**
Larson, M., & Slokom, M. (2019). Up close, but not too personal: Hypotargeting for recommender systems. In O. S. Shalom, D. Jannach, & I. Guy (Eds.), *ImpactRS 2019 Impact of Recommender Systems 2019: Proceedings of the 1st Workshop on the Impact of Recommender Systems co-located with 13th ACM Conference on Recommender Systems (ACM RecSys 2019)* (pp. 1-2). (CEUR Workshop Proceedings; Vol. 2462). CEUR-WS. http://ceur-ws.org/Vol-2462/

# Up Close, but not too Personal: Hypotargeting for Recommender Systems[*]

Martha Larson
Radboud University and TU Delft
Netherlands
m.larson@cs.ru.nl

Manel Slokom
TU Delft
Netherlands
m.slokom@tudelft.nl

## ABSTRACT

Hypotargeting for recommender systems (hyporec) is the idea of controlling the number of unique lists of items that a recommender system can recommend to users during a given time period. The main advantage of hyporec is *oversight*. If a recommender system offers only a finite number of unique lists, then it becomes feasible for a person without technological knowledge to audit the recommender system. Oversight makes it possible to spot filter bubbles or cases in which users are being bombarded with divisive content. We argue that hyporec is actually not so far from many existing recommender system ideas, and that with further research hyporec systems could be capable of making good tradeoffs between the number of unique lists, rate of list renewal (which controls coverage), and conventional evaluation metrics for user satisfaction.

## CCS CONCEPTS

• **Information systems → Recommender systems**.

## KEYWORDS

position paper, oversight, personalization



## 1 INTRODUCTION

In this position paper, we present the case for *hyporec*, hypotargeing for recommender systems. Hyporec uses parameters to control the impact of a recommender system on the user population. Specifically, it controls the number of unique experiences that the system offers to users in the user population. Formally, we define hyporec to be any recommender algorithm that includes two additional parameters $\mathcal{L}$ and $\mathcal{T}$. $\mathcal{L}$ is the number of unique lists of recommended items that a recommender system is allowed to recommend. $\mathcal{T}$ is the time period after which $\mathcal{L}$ is allowed to change. We use the word 'experience' in the context of hyporec to refer to a specific list of recommendations (note, the difference with 'user experience', which encompasses many other factors). Depending on the recommender, the 'experience' might not actually be a list of items, and the members of $\mathcal{L}$ may also be sets or sequences.

The consequences of introducing control by adding parameters $\mathcal{L}$ and $\mathcal{T}$ are straightforward. Let $\mathcal{U}$ be the total number of users in the population. If $\mathcal{L}$ is set $\geqslant \mathcal{U}$ and if $\mathcal{T}$ is very large we have a conventional recommender system, which admits the possibility of every user receiving a unique recommendation list. We emphasize *possibility*, since conventional recommender systems do not necessarily generate a unique recommendation list for each user. So, in fact a hyporec recommender system may reduce to a conventional recommender system when $\mathcal{L}$ is somewhat less than $\mathcal{U}$.

As $\mathcal{L}$ becomes smaller and takes on values $\ll \mathcal{U}$, it starts to constrain the recommender. Given a target user, $u$, and a timepoint, $t$, the system is forced to pick the best list for $u$ from the lists that are available in $\mathcal{L}$ at $t$. Like a conventional recommender, the list is picked based on the user profile and the context. If $\mathcal{L}$ is very small, it is possible that many users receive item lists containing no items relevant for them. However, a hyporec recommender system is intended to be optimized to find a more suitable operating point than this one. The purpose of hyporec is to allow specification of the number of unique experiences (by setting $\mathcal{L}$) that are served to the user population during a given time interval of length $\mathcal{T}$.

If $\mathcal{L}$ is too small (i.e., the choice of lists is very limited) and simultaneously $\mathcal{T}$ is very large (i.e., that choice is nearly never allowed to change) the result is poor catalogue coverage. There could be items in the system with no chance of being exposed to users. Again, a hyporec recommender system is not intended to run at such an operating point.

The main driving motivation between the hyporec idea is to provide a recommender system with extra parameters that can be set in order to gain additional control on the overall impact of the system on users. Specifically, we are interested in improving the insight into the functioning of the recommender system that is available to people without technical knowledge. Such insight enables oversight by a third party, including both external auditors and the users themselves. The motivation supporting our position that hyporec is interesting and important for the recsys community to research will be discussed in Section 3, after we have discussed the relation of hyporec to other recommenders.

## 2 HYPOREC VS. OTHER RECOMMENDERS

Discussing some simple theoretical properties allow us to highlight the relation between hyporec and other recommenders. A first glance, adding parameters $\mathcal{L}$ and $\mathcal{T}$ appear to result in a system that is strange and unfamiliar. However, upon closer consideration it is more familiar that one would think.

### 2.1 Popularity-based recommendation

If we set $\mathcal{L} = 1$ and we chose $l \in \mathcal{L}$ to be a list of the top-most popular items, then we have a familiar popularity-based recommender. This recommender satisfies a large number of users with only a single list at its disposal. We could then increase $\mathcal{L}$ and strategically compose lists of popular items. This would allow us to build a system that would provide a very large number of users (nearly everyone) with a list containing a minimal number of recommendations they would find relevant.


Presented at the ImpactRS workshop held in conjunction with the 13th ACM Conference on Recommender Systems (RecSys), 2019, in Copenhagen, Denmark.

Note that it is not necessary that hyporec fills the $\mathcal{L}$ available lists with popular items. The popularity example simply provides a clear demonstration that hyporec is theoretically capable of providing relevant recommendations to a large number of users. We envision that actual hyporec will be more sophisticated, and also balance $\mathcal{L}$ and conventional metrics.

## 2.2 Group recommendation

A group recommender generates recommendations for a group of users, rather than for individual users. The goal is to satisfy some balance or combination of the preferences of the users in that group. In group recommendation, the groups are defined in advance of the recommendation (e.g., the group is several friends who want to watch movie together). Hyporec recommender systems also offer the same list of items to a group of people. However, there are two critical differences. The first is that the group is not defined in advance, but is constituted as part of the recommender algorithm. The second is that the relevance of the recommendation list is assessed with respect to the individual group members.

## 2.3 Diversification

Recommender systems that pursue diversification work to ensure that users receive a wide view on the content available, which helps them to avoid the negative effects of filter bubbles, and also develop new interests. If we want to serve the most possible users with the limited $\mathcal{L}$ at our disposal in hyporec, we must be creative about constituting lists. Lists can be relevant to different users in different ways. For example, a good choice for a list $l \in \mathcal{L}$ is one in which the items relevant to $u_b$ are actually diverse items with respect to $u_a$. Hyporec is different from diversification in that it may be the case that an optimal list within hyporec is an interleaving of items exclusively relevant for either $u_a$ or $u_b$.

# 3 IMPORTANCE OF HYPOREC

Our position that recommender systems need to explore two new hyporec parameters $\mathcal{L}$ and $\mathcal{T}$. In this section, we explain why these two, rather unassuming, parameters are so important.

## 3.1 Oversight

As we limit the number of unique lists that a recommender can recommend, we gain the possibility of *oversight*: it becomes feasible to audit the experiences that a recommender system is offering to its users at a given moment by simply paging through them. Hyporec does not prevent filter bubbles out of the box, but if the number of possible recommendation lists is limited, it becomes feasible for users to actually *compare* their experience with the experience of other users, and determine for themselves whether or not they are in a filter bubble. No technical background (e.g., understanding of recommender system metrics) is necessary to do this. Limiting the number of views will make it possible to regulate recommendation. Regulators can also page through recommendations, and look for worrisome system behavior. Hyporec systems will be able to offer a list archive, much like Facebook has opened an advertisement library[1] where it is possible for users to explore which ads are being served by Facebook, giving them an overall view of Facebook advertising practices.

## 3.2 Avoiding overfitting users

Recommender systems attempt to learn users item preferences, but may inadvertently reinforce content (especially divisive content) which unduly influences users. Mainstream news is currently highlighting the dangers of such systems[2]. Another example is that a recommender system might continue to recommend movies involving a certain type of personal tragedy, that a user feels compelled to watch, but which spirals them into a state of distress. With hyporec, a list must serve for multiple users, so it is not possible to bombard a user with lists containing exclusively type of content that s/he personally cannot pull away from. Should a problem arise with a recommender system, the fact that there are a finite number of lists (see "Oversight" above) makes it much easier to find than in current recommender systems.

## 3.3 Social connection

Recommender systems which admit the possibility of providing each user with a unique list of recommendations are socially isolating. For example, news recommenders make it frustrating to attempt to determine if there is actually more sexual assault going on in the world, or if I am simply seeing more stories about sexual assault because I keep on reading them and the recommender system is learning my reading pattern. If there is absolutely no one else in the world who shares the same experience of a recommender system, there is no one else who I can confide in to explain my growing distress. No one will believe me. This effect is important because the items of the list/set/sequence impact the user as a set (which is why we use the word 'experience'). Even if other users see the same individual items, but if they do not see the same set, they cannot communicate with each other about what they are seeing. A variant of hyporec is not to control the number of lists by $\mathcal{L}$, but rather by $\mathcal{M}$, a parameter that specifies the minimal number of users who must be recommended any given list.

## 3.4 Privacy

The harmful effects of behaviorial advertising (discrimination, manipulation, filter bubbles) are becoming more widely understood, leading to calls to return to using recommendations based on context information to replace personalization[3]. Restricting the number of recommended lists to $\mathcal{L}$ also limits the amount of personal information needed to make effective recommendations, because a good match is enough, and an exact match is no longer necessary.

# 4 OUTLOOK

This vision paper is a call to the recsys community to investigate hypotargeting. It opens interesting research questions (such as how to run a stream recommender so at the end $\mathcal{T}$ having used $\mathcal{L}$ unique lists). Hyporec recommenders are not a silver bullet solution: just like a conventional recommenders they are capable of producing undesirable effects. The advantage is that harmful effects are easier to address because they are less likely to remain hidden.

[1] https://www.facebook.com/ads/library

[2] https://www.nytimes.com/interactive/2019/06/08/technology/youtube-radical.html

[3] https://www.nytimes.com/2019/06/19/opinion/facebook-google-privacy.html